



## Comparison of openfoam and ellipsys for modelling the wind ressources in complex terrain

Réthoré, Pierre-Elouan; Martinez, Benjamin ; Cavar, Dalibor; Zahle, Frederik; Berg, Jacob; Sørensen, Niels N.

*Publication date:*  
2012

*Document Version*  
Publisher's PDF, also known as Version of record

[Link back to DTU Orbit](#)

*Citation (APA):*  
Réthoré, P-E., Martinez, B., Cavar, D., Zahle, F., Berg, J., & Sørensen, N. N. (2012). *Comparison of openfoam and ellipsys for modelling the wind ressources in complex terrain*. Poster session presented at EWEC 2012 - European Wind Energy Conference & Exhibition, Copenhagen, Denmark.

---

### General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.



## Abstract

The flow solvers OpenFOAM<sup>6</sup> and EllipSys<sup>4,5</sup> are compared on the atmospheric flow over terrain test cases of Askervein. Both solvers are run with the steady state Reynold Averaged Navier Stokes (RANS) k-epsilon turbulence model. One of the main modelling differences in-between the two solver is the law of the wall function. OpenFOAM uses a Nikuradse's<sup>2</sup> sand roughness length model, while EllipSys uses Richard & Hoxey's<sup>1</sup> surface roughness.

It is found that Nikuradse's<sup>2</sup> model introduces an error dependent to the first cell height. To mitigate this error the first cell should be at least ten times larger than the surface roughness. It is nonetheless possible to obtain very close results between EllipSys and OpenFOAM.

Two meshing strategies are investigated, using HypGrid<sup>3</sup> and SnappyHexMesh. HypGrid is found to give consistently smaller meshes than SnappyHexMesh. Both mesh generator produce meshes that can perform accurately on the Askervein test case. SnappyHexMesh is however found to be difficult to use on very complex terrains.

OpenFOAM is found to be consistently about 10 times slower than EllipSys on similar mesh sizes. However, OpenFOAM could potentially be tuned-up with a deeper knowledge of the flow solver inputs.

## Objectives

OpenFOAM is a free and open source flow solver that has recently attracted a lot of attention both in the wind industry and research community. However the cost of a flow solver is not only the license cost but also the human cost (meshing time, and tuning the solver) and the computational cost (meshing and running the simulation). The purpose of this study is to compare it with EllipSys on a complex terrain case, focusing on those two aspects as well as on the solvers accuracy.

The mesh generator of OpenFOAM, SnappyHexMesh is used and compared with the mesh generator of EllipSys, HypGrid. HypGrid meshes can also be used with OpenFOAM.

## Methods

EllipSys<sup>4,5</sup> and OpenFOAM<sup>6</sup> are run using the same turbulence model (RANS steady state k-epsilon), using a QUICK scheme and a SIMPLE pressure solver and a multigrid solver.

The two flow solvers use two different wall functions. The Richard & Hoxey's formulation, used by EllipSys, is working best for a first cell height of  $y_p < 0.5 y_0$ .

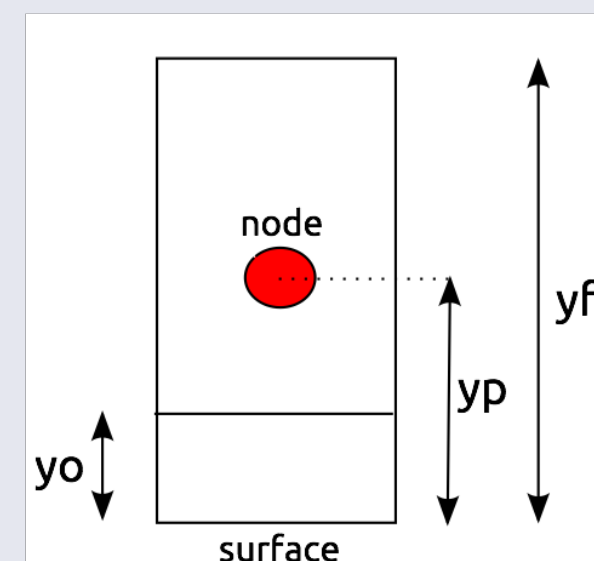
The Nikuradse's formulation, by OpenFOAM, is inconsistent: the inlet formula, based on the roughness length is different from the log-law obtained using the wall function, which is based on the  $k_s$  formulation. To mitigate the error, the first cell height should be roughly  $y_p > 10 y_0$ .

$$\frac{u_p}{u_*} = \frac{1}{\kappa} \ln\left(\frac{y_p + y_0}{y_0}\right)$$

EllipSys: Richard & Hoxey<sup>1</sup>

$$\frac{u_p}{u_*} = \frac{1}{\kappa} \ln\left(\frac{E y_p}{C_s k_s}\right)$$

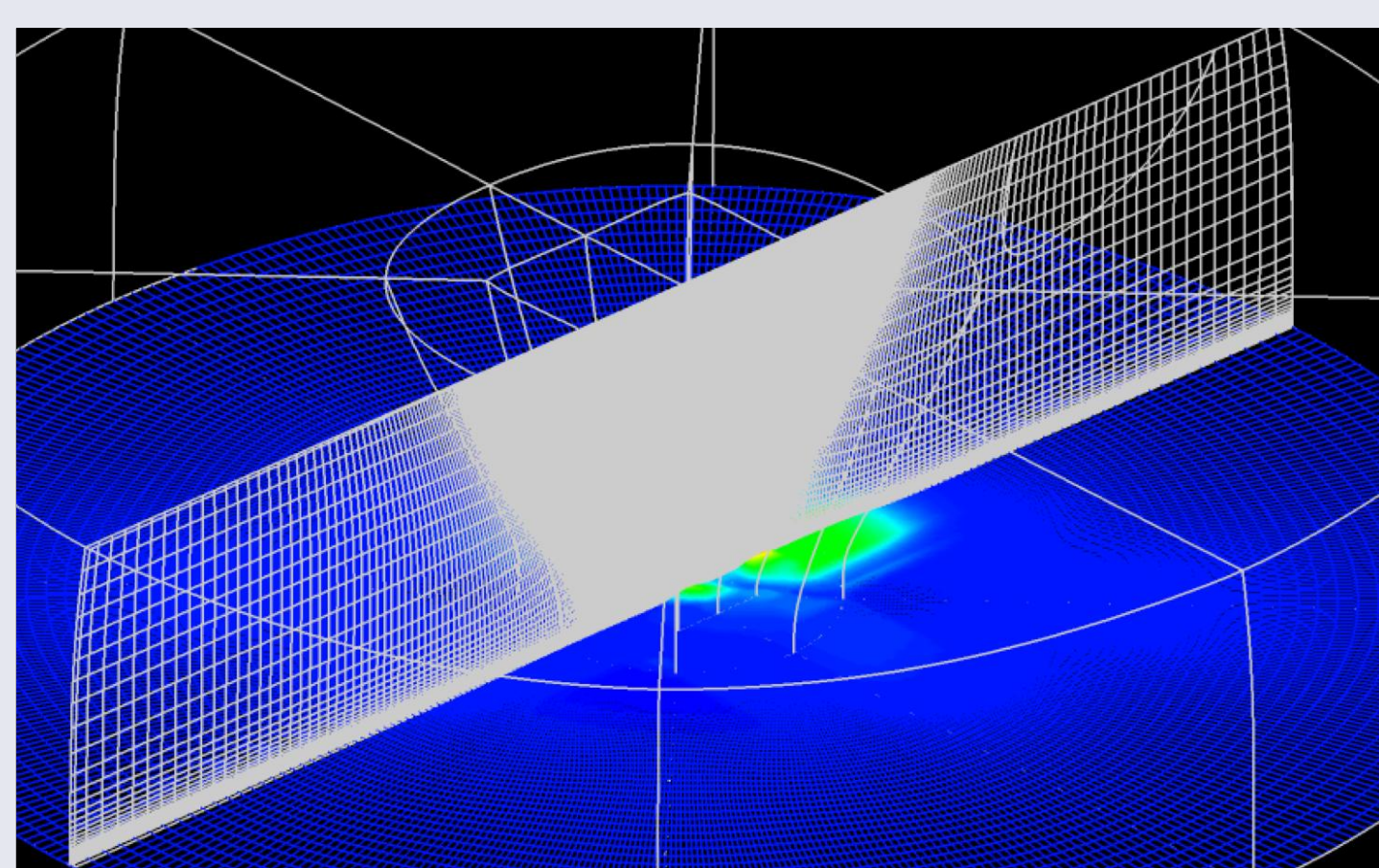
OpenFOAM: Nikuradse<sup>2</sup>



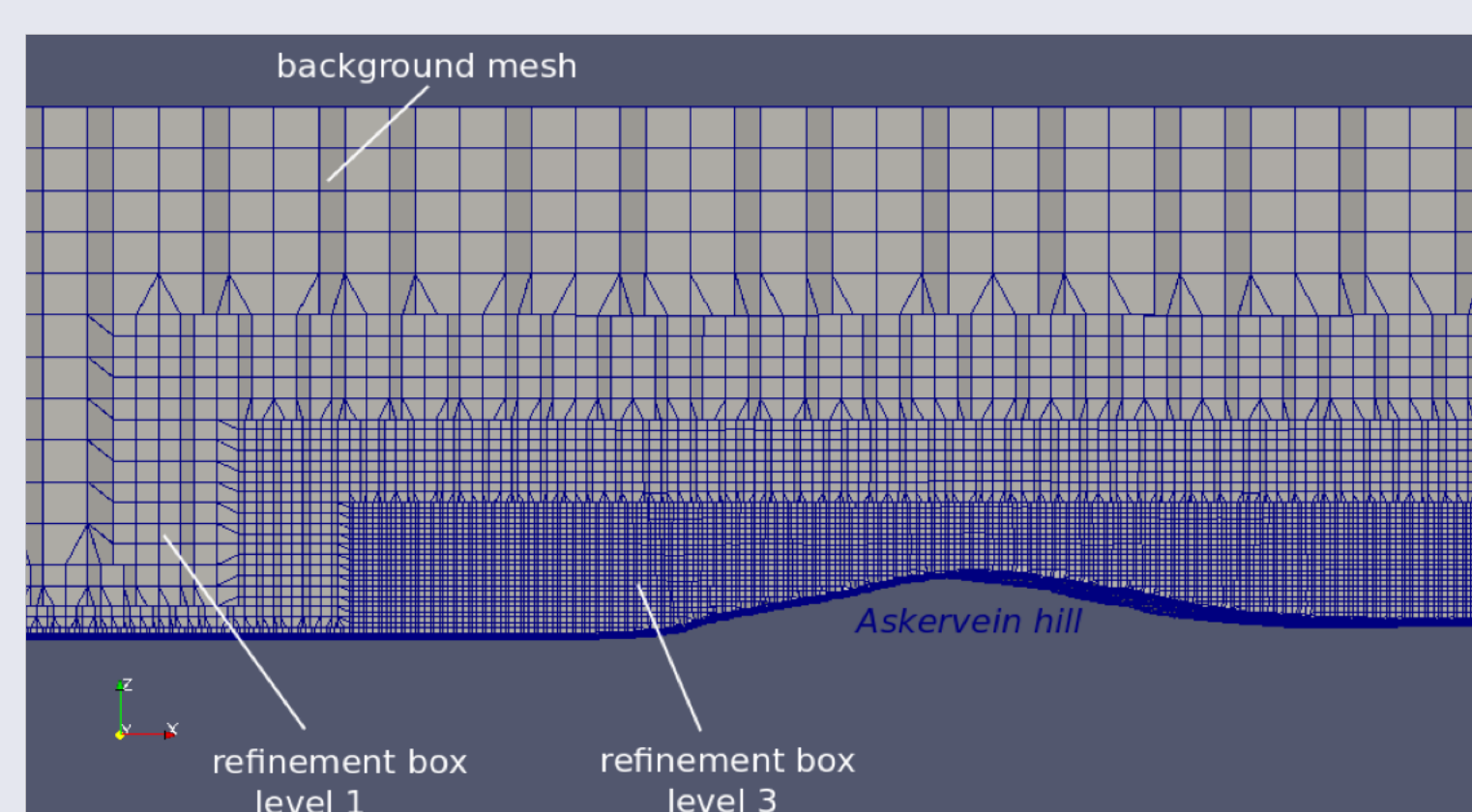
EllipSys uses structured meshes, while OpenFOAM can use both structured and unstructured meshes.

OpenFOAM comes with a meshing tool called SnappyHexMesh. One restriction of OpenFOAM meshes is that the cells aspect ratio should be close to one. The larger the aspect ratio, the slower OpenFOAM converges, and the larger is the numerical error.

EllipSys does not have restrictions on cells aspect ratio. It uses an hyperbolic mesh generator that takes advantages of this feature and creates smaller meshes.



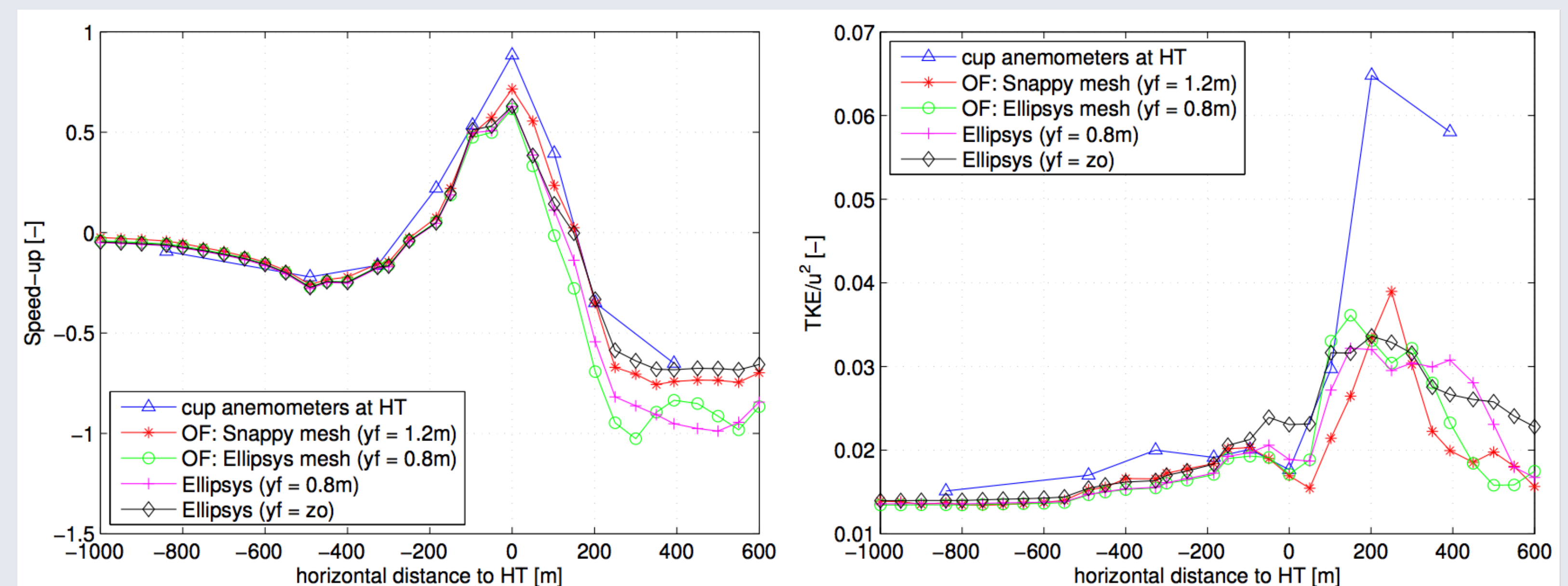
Askervein meshed with HypGrid



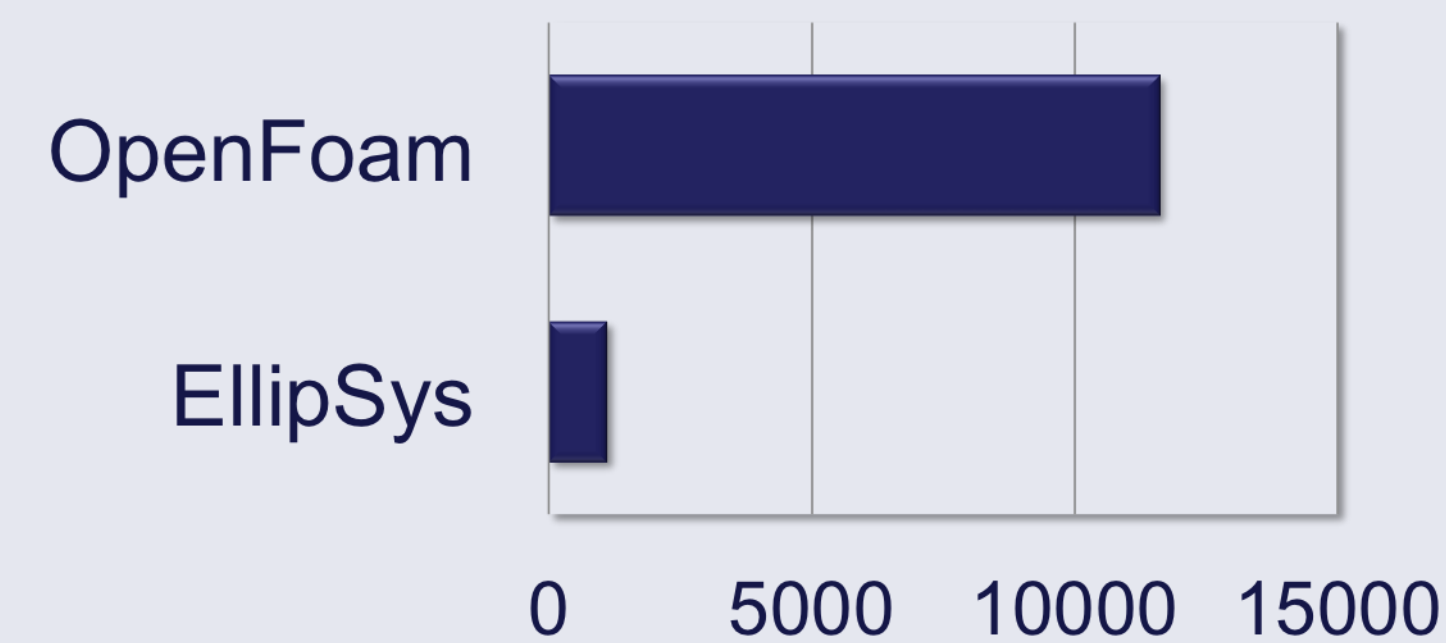
Askervein meshed with SnappyHexMesh

## Results

When using the meshes generated with HypGrid, OpenFOAM and EllipSys predict very similar speed-up's over the hill top. Only small differences are observed in the first few meters over terrain. They could be related to difference in grid refinement and to the wall function inconsistency in OpenFOAM. SnappyHexMesh, predicts a higher speed-up. The reason for this remains uncertain, although it should be mentioned that the refinement of the near surface isn't done appropriately (due to the restriction of the mesher) and is yielding more inaccuracy. The turbulent kinetic energy (TKE) is well predicted by all simulations, except in the hill recirculation region. This result is in agreement with many studies<sup>4</sup>.

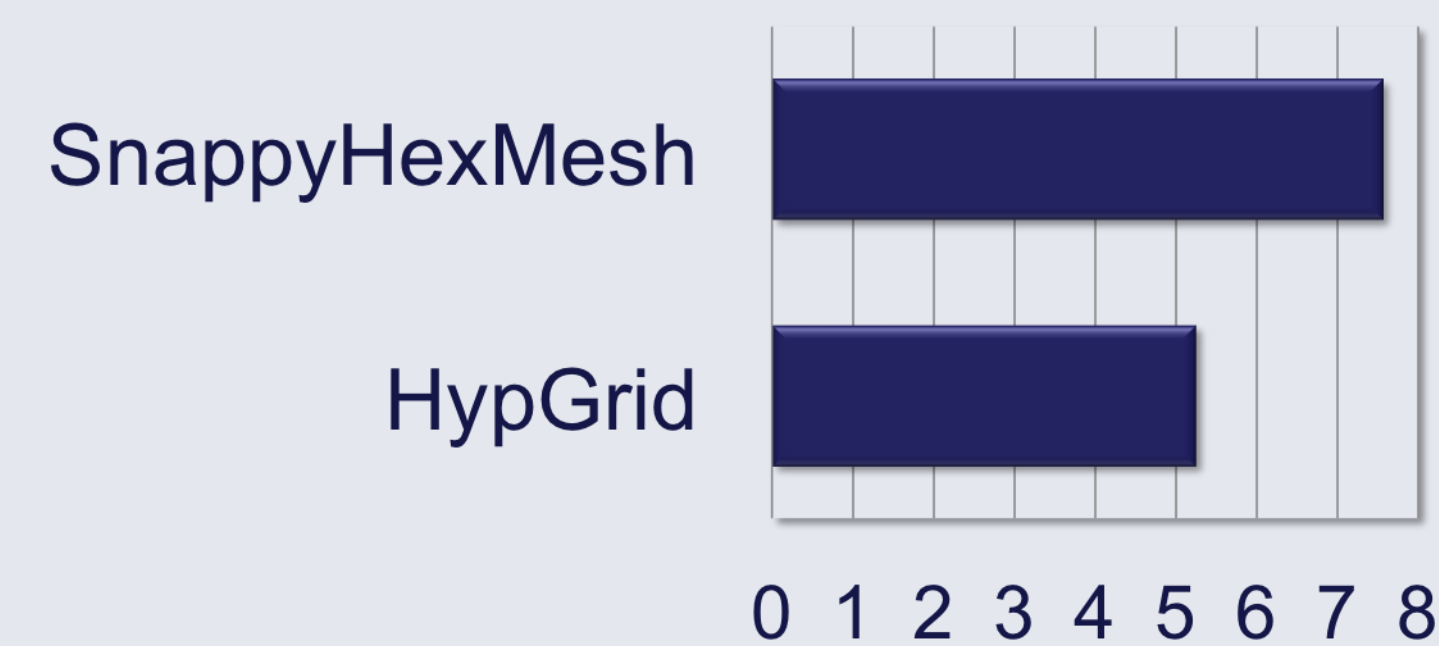


### Simulation time [s]



The OpenFOAM is approximately ten times slower than EllipSys. The authors are quite positive that it can be tuned-up changing some inputs of the GAMG pressure solver. The lack of information regarding this solver made it difficult to set the most appropriate inputs.

### Number of cells [1E6]



Terrain mesh generation is difficult and computationally expensive with SnappyHex-Mesh and even though some cells could have been saved by using unstructured meshes, it creates relatively larger meshes than HypGrid for the same level of refinement.

## Conclusions

OpenFOAM has been successfully validated on the Askervein test case, and compared with EllipSys. OpenFOAM and EllipSys are found to perform equally well when used with their respective mesh requirements.

Even though OpenFOAM is coded with Nikuradse law of the wall for rough surfaces, which is known to be inconsistent and problematic in many CFD packages, inaccuracy is only observed in the first few cells (vertical direction) of the computational domain.

In order to overcome this law of the wall inconsistency issue, it is suggested as a future work, to implement new wall boundary conditions based on Richard and Hoxey's profiles in OpenFOAM.

The snappyHexMesh utility has proven to have its potential when observing its fairly good agreement with respect to measurements and to the simulations performed with EllipSys and OpenFOAM on the HypGrid mesh. Nevertheless, the utility has its limits, as many problems are encountered when meshing the relatively shallow Askervein hill. Moreover, it is found difficult to maintain the total amount of cells to a reasonably low value because of the maximum aspect ratio limitation of the background mesh.

## Acknowledgement

This work was partly funded by the DSF "Center for Computational Wind Turbine Aerodynamics and Atmospheric Turbulence" project c.no.: 2104-09-0026.

## References

- [1] P.J. Richards and R.P Hoxey. *J. of Wind Eng.* 1993
- [2] J. Nikuradse. *NACA Tech. Mem.* 1937
- [3] N.N. Sørensen. "HypGrid2d". *Tech rep. Risø.* 1998.
- [4] N.N. Sørensen. *PhD Thesis. Risø.* 1995.
- [5] J.A. Michelsen. Basis3d. *Tech. Rep. AFM. DTU.* 1992
- [6] <http://openfoam.org>